AMENDMENT TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application.

Listing of Claims:

1. (currently amended)   A system for managing coherent data access through multiple nodes, comprising:

a first data processing system forming a first node, in which the first data processing system includes a first bridge, a first interface and a memory that is local to the first node, wherein the first node supports ~~coherent~~ packet traffic for transfer of packets, coherent traffic to access local and remote memory and non-coherent traffic ~~and~~ to communicate with input/output (I/O) circuitry, in which the memory stores cacheable data having coherency; and

a second data processing system forming a second node that supports packet traffic, coherent traffic and non-coherent traffic, in which the second data processing system includes a second bridge and a second interface, the first and second interfaces coupling the first node to the second node for transfer of packet traffic, coherent traffic and non-coherent traffic, wherein when the second node receives a ~~request~~ packet from an external source that is to access a coherent fabric of the memory, the second bridge identifies the first node as a remote node and ~~transfers the request~~ converts the packet as an uncacheable data access request to the first node so that the uncacheable data access request does not access a coherent fabric in the second node, and when the first bridge receives the uncacheable data access request, the first bridge identifies the uncacheable data access request as a local access request to the memory ~~as a local access~~ in the first node and processes the uncacheable data access request from the second node as a coherent data access request in the first node to access the coherent fabric of the memory in the first node.

2. (canceled)

3.   (currently amended) The system of claim 1, wherein the <u>uncacheable data access</u> request <u>associated with the packet </u>from the external source is a ~~read~~ <u>store </u>or a write request to access the memory.

4-5. (canceled)

6.   (currently amended) The system of claim 1, wherein the <u>uncacheable data </u>access <u>request </u>by the second bridge ~~writes data to a location of the memory and a subsequent access by an agent to read the data in the memory conform to a producer-consumer protocol, wherein the second bridge corresponds to a producer and the agent corresponds to a consumer of the~~ <u>follows a </u>producer-consumer protocol.

7.   (currently amended) The system of claim 6, wherein ~~the data written by the second bridge to the memory comprises~~ a payload and a flag<u> are written when following the producer-consumer protocol</u>.

8-9. (canceled)

10. (currently amended) A method for managing coherent data access through multiple nodes, comprising:

establishing a cacheable coherent memory space in a local memory of a first data processing system that forms a first node that supports <u>packet traffic for transfer of packets, </u>coherent <u>traffic to access local and remote memory </u>and non-coherent traffic<u> to communicate with input/output (I/O) circuitry</u>, and in which the first data processing system also includes a first bridge and a first interface;

receiving at a second node a ~~request~~ <u>packet </u>from an external source to access a coherent fabric of the memory<u> in the first node</u>, wherein the second node supports <u>packet </u>traffic, <u>coherent <u>traffic </u>and non-coherent traffic and is formed of a second data processing system that includes a second bridge and a second interface;

identifying in the second node that the memory <u>to be accessed </u>is located in a remote node;

converting ~~the request~~ the packet in the second bridge as an uncacheable data access request to access the memory ~~as an uncacheable access request to~~ in the first node ~~in the second bridge~~, so that the uncacheable data access request does not access a coherent fabric in the second node;

transferring by the second bridge the uncacheable data access request to the first node;

receiving by the first bridge the uncacheable data access request from the second node through coupling between the first and second interfaces;

identifying the uncacheable data access request as a local access to the memory in the first node;

processing the uncacheable data access request from the second node as a coherent data access to access the coherent fabric of the memory in the first node; and

accessing the coherent fabric of the memory in the first node in response to ~~the request~~ receiving the packet from the external source ~~to access the memory~~, but ~~in which the access is processed~~ without accessing the coherent fabric in the second node.

11. (canceled)

12. (currently amended) The method of claim 10, wherein the uncacheable data access request associated with the packet from the external source is a ~~read access~~ store or a write request to access ~~to~~ the memory.

13-14. (canceled)

15. (currently amended) The method of claim 10, wherein the uncacheable data access request by the second bridge ~~writes data to a location of the memory and a subsequent access by an agent to read the data in the memory conform to a producer-consumer protocol, wherein the second bridge corresponds to a producer and the agent corresponds to a consumer of the~~ follows a producer-consumer protocol.

16. (currently amended) The method of claim 15, wherein ~~the data written by the second bridge to the memory comprises~~ a payload and a flag <u>are written when following the producer-consumer protocol</u>.

17-18. (canceled)